

HMRF-EM-image: Implementation of the Hidden Markov Random Field Model and its Expectation-Maximization Algorithm

Quan Wang

Signal Analysis and Machine Perception Laboratory
Electrical, Computer, and Systems Engineering
Rensselaer Polytechnic Institute

wangq10@rpi.edu

Abstract

In this project¹, we study the hidden Markov random field (HMRF) model and its expectation-maximization (EM) algorithm. We implement a MATLAB toolbox named HMRF-EM-image for 2D image segmentation using the HMRF-EM framework². This toolbox also implements edge-prior-preserving image segmentation, and can be easily reconfigured for other problems, such as 3D image segmentation.

1. Introduction

Markov random fields (MRFs) have been widely used for computer vision problems, such as image segmentation [4], surface reconstruction [3] and depth inference [2].

The HMRF-EM framework was first proposed for segmentation of brain MR images [5]. Given an image $\mathbf{y} = (y_1, \dots, y_N)$ where each y_i is the intensity of a pixel, we want to infer a configuration of labels $\mathbf{x} = (x_1, \dots, x_N)$ where $x_i \in L$ and L is the set of all possible labels. In a binary segmentation problem, $L = \{0, 1\}$. According to the MAP criterion, we seek the labeling \mathbf{x}^* which satisfies

$$\mathbf{x}^* = \underset{\mathbf{x}}{\operatorname{argmax}} \{P(\mathbf{y}|\mathbf{x}, \Theta)P(\mathbf{x})\}. \quad (1)$$

The prior probability $P(\mathbf{x})$ is a Gibbs distribution, and the joint likelihood probability is

$$\begin{aligned} P(\mathbf{y}|\mathbf{x}, \Theta) &= \prod_i P(y_i|\mathbf{x}, \Theta) \\ &= \prod_i P(y_i|x_i, \theta_{x_i}), \end{aligned} \quad (2)$$

where $P(y_i|x_i, \theta_{x_i})$ is a Gaussian distribution with parameters $\theta_{x_i} = (\mu_{x_i}, \sigma_{x_i})$. $\Theta = \{\theta_l | l \in L\}$ is the parameter set, which is obtained by the EM algorithm.

2. EM Algorithm

We use the EM algorithm to estimate the parameter set $\Theta = \{\theta_l | l \in L\}$. We describe the EM algorithm by the following:

1. *Start*: Assume we have an initial parameter set $\Theta^{(0)}$.
2. *E-step*: At the t th iteration, we have $\Theta^{(t)}$, and we calculate the conditional expectation:

$$\begin{aligned} Q(\Theta|\Theta^{(t)}) &= E \left[\ln P(\mathbf{x}, \mathbf{y}|\Theta) | \mathbf{y}, \Theta^{(t)} \right] \\ &= \sum_{\mathbf{x} \in \chi} P(\mathbf{x}|\mathbf{y}, \Theta^{(t)}) \ln P(\mathbf{x}, \mathbf{y}|\Theta), \end{aligned} \quad (3)$$

where χ is the set of all possible configurations of labels.

3. *M-step*: Now maximize $Q(\Theta|\Theta^{(t)})$ to obtain the next estimate:

$$\Theta^{(t+1)} = \underset{\Theta}{\operatorname{argmax}} Q(\Theta|\Theta^{(t)}). \quad (4)$$

Then let $\Theta^{(t+1)} \rightarrow \Theta^{(t)}$ and repeat from the E-step.

Let $G(z; \theta_l)$ denote a Gaussian distribution function with parameters $\theta_l = (\mu_l, \sigma_l)$:

$$G(z; \theta_l) = \frac{1}{\sqrt{2\pi}\sigma_l} \exp \left(-\frac{(z - \mu_l)^2}{2\sigma_l^2} \right). \quad (5)$$

We assume that the prior probability can be written as

$$P(\mathbf{x}) = \frac{1}{Z} \exp(-U(\mathbf{x})), \quad (6)$$

¹This work originally appears as the final project of Prof. Birsen Yazici's course *Detection and Estimation Theory* at RPI.

²This toolbox can be downloaded at the author's homepage <http://homepages.rpi.edu/wangq10>.

where $U(\mathbf{x})$ is the prior energy function. We also assume that

$$\begin{aligned} P(\mathbf{y}|\mathbf{x}, \Theta) &= \prod_i P(y_i|x_i, \theta_{x_i}) \\ &= \prod_i G(y_i; \theta_{x_i}) \\ &= \frac{1}{Z'} \exp(-U(\mathbf{y}|\mathbf{x})). \end{aligned} \quad (7)$$

With these assumptions, the HMRF-EM algorithm is given below:

1. Start with initial parameter set $\Theta^{(0)}$.
2. Calculate the likelihood distribution $P^{(t)}(y_i|x_i, \theta_{x_i})$.
3. Using current parameter set $\Theta^{(t)}$ to estimate the labels by MAP estimation:

$$\begin{aligned} \mathbf{x}^{(t)} &= \operatorname{argmax}_{\mathbf{x} \in \chi} \{P(\mathbf{y}|\mathbf{x}, \Theta^{(t)})P(\mathbf{x})\} \\ &= \operatorname{argmin}_{\mathbf{x} \in \chi} \{U(\mathbf{y}|\mathbf{x}, \Theta^{(t)}) + U(\mathbf{x})\}. \end{aligned} \quad (8)$$

The algorithm for the MAP estimation is discussed in Section 3.

4. Calculate the posterior distribution for all $l \in L$ and all pixels y_i :

$$P^{(t)}(l|y_i) = \frac{G(y_i; \theta_l)P(l|x_{N_i}^{(t)})}{P^{(t)}(y_i)}, \quad (9)$$

where $x_{N_i}^{(t)}$ is the neighborhood configuration of $x_i^{(t)}$, and

$$P^{(t)}(y_i) = \sum_{l \in L} G(y_i; \theta_l)P(l|x_{N_i}^{(t)}). \quad (10)$$

Note here we have

$$P(l|x_{N_i}^{(t)}) = \frac{1}{Z} \exp\left(-\sum_{j \in N_i} V_c(l, x_j^{(t)})\right). \quad (11)$$

5. Use $P^{(t)}(l|y_i)$ to update the parameters:

$$\mu_l^{(t+1)} = \frac{\sum_i P^{(t)}(l|y_i)y_i}{\sum_i P^{(t)}(l|y_i)}, \quad (12)$$

$$(\sigma_l^{(t+1)})^2 = \frac{\sum_i P^{(t)}(l|y_i)(y_i - \mu_l^{(t+1)})^2}{\sum_i P^{(t)}(l|y_i)}. \quad (13)$$

3. MAP Estimation

In the EM algorithm, we need to solve for \mathbf{x}^* that minimizes the total posterior energy

$$\mathbf{x}^* = \operatorname{argmin}_{\mathbf{x} \in \chi} \{U(\mathbf{y}|\mathbf{x}, \Theta) + U(\mathbf{x})\} \quad (14)$$

with given \mathbf{y} and Θ , where the likelihood energy is

$$\begin{aligned} U(\mathbf{y}|\mathbf{x}, \Theta) &= \sum_i U(y_i|x_i, \Theta) \\ &= \sum_i \left[\frac{(y_i - \mu_{x_i})^2}{2\sigma_{x_i}^2} + \ln \sigma_{x_i} \right]. \end{aligned} \quad (15)$$

The prior energy function $U(\mathbf{x})$ has the form

$$U(\mathbf{x}) = \sum_{c \in C} V_c(\mathbf{x}), \quad (16)$$

where $V_c(\mathbf{x})$ is the clique potential and C is the set of all possible cliques.

In the image domain, we assume that one pixel has at most 4 neighbors: the pixels in its 4-neighborhood. Then the clique potential is defined on pairs of neighboring pixels:

$$V_c(x_i, x_j) = \frac{1}{2}(1 - I_{x_i, x_j}), \quad (17)$$

where

$$I_{x_i, x_j} = \begin{cases} 0 & \text{if } x_i \neq x_j \\ 1 & \text{if } x_i = x_j \end{cases}. \quad (18)$$

We have developed an iterative algorithm to solve (14):

1. To start with, we have an initial estimate $\mathbf{x}^{(0)}$, which is from the previous loop of the EM algorithm.
2. Provided $\mathbf{x}^{(k)}$, for all $1 \leq i \leq N$, we find
$$x_i^{(k+1)} = \operatorname{argmin}_{l \in L} \{U(y_i|l) + \sum_{j \in N_i} V_c(l, x_j^{(k)})\}. \quad (19)$$
3. Repeat step 2 until $U(\mathbf{y}|\mathbf{x}, \Theta) + U(\mathbf{x})$ converges or a maximum k is achieved.

4. Edge-Prior-Preserving Image Segmentation

To use HMRF-EM framework for image segmentation, first we generate an initial segmentation using k-means clustering on the gray-level intensities of pixels. The initial segmentation provides the initial labels $\mathbf{x}^{(0)}$ for the MAP algorithm, and the initial parameters $\Theta^{(0)}$ for the EM algorithm. Then we run the EM algorithm, and the resulting label configuration \mathbf{x} will be a refined segmentation result.

Now we would like our segmentation to preserve the edges obtained by some edge detection algorithm. Assume we have a binary edge map \mathbf{z} , where $z_i = 1$ if the i th pixel is on an edge, and $z_i = 0$ if not. Then we modify (19) to

$$x_i^{(k+1)} = \operatorname{argmin}_{l \in L} \{U(y_i|l) + \sum_{j \in N_i, z_j=0} V_c(l, x_j^{(k)})\}. \quad (20)$$

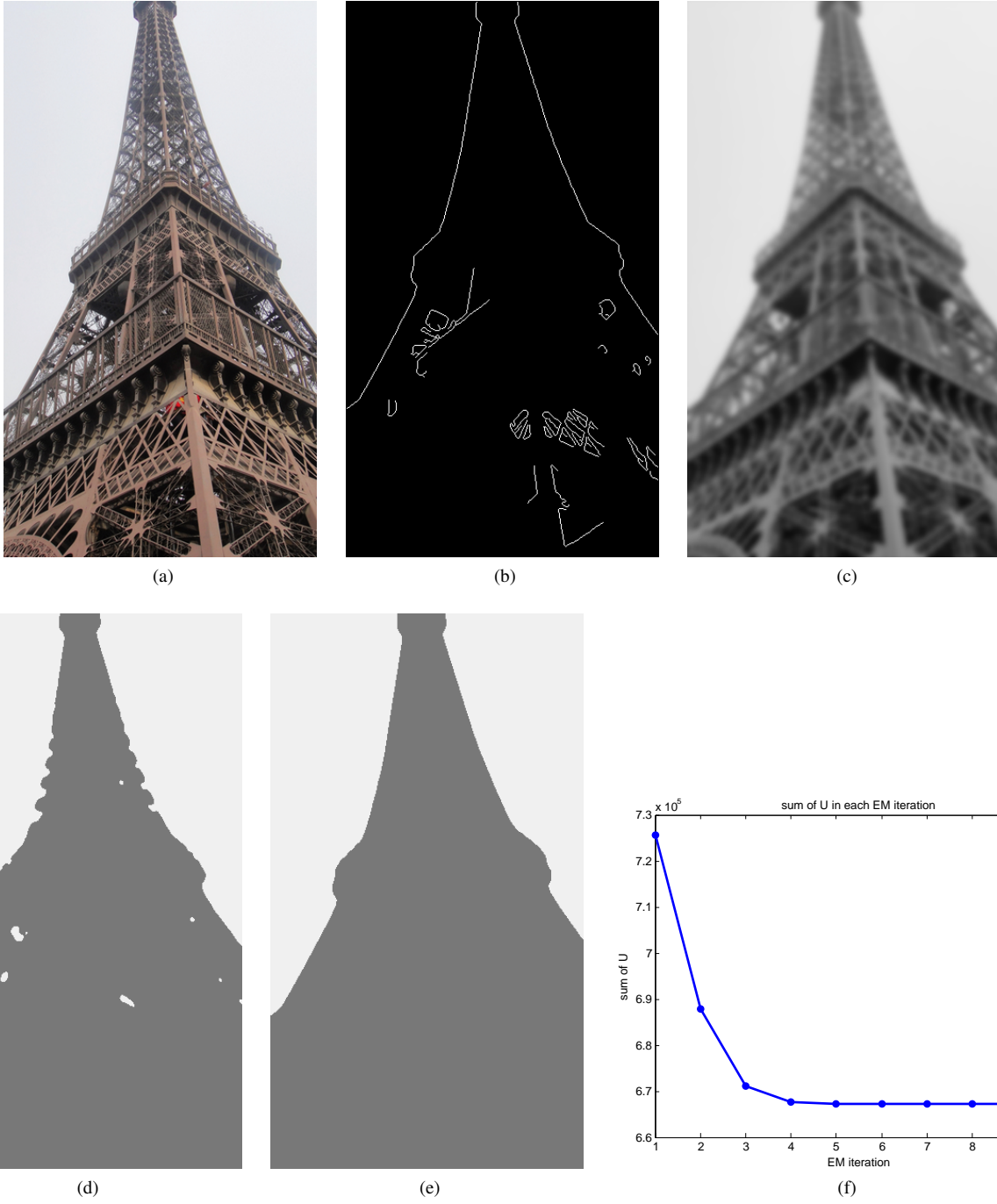


Figure 1: Edge-prior-preserving image segmentation results. (a) Original image. (b) Canny edges. (c) Gaussian blurred image. (d) Initial labels obtained by k-means, where $k = 2$. (e) Final labels obtained by HMRf-EM algorithm. (f) Total posterior energy in each iteration of the EM algorithm.

5. Experiment Results

We run our HMRf-EM edge-prior-preserving segmentation algorithm on example images. The binary edge map \mathbf{z}

is obtained by performing Canny edge detection [1] on the original image, and the observation \mathbf{y} is obtained by performing Gaussian blur on the original image. Some results are shown in Figure 1. We can see that the initial labels

File	Type	Usage
demo.m	Runnable script	A demo showing how to use the toolbox. Users can run this file directly.
image_kmeans.m	Function	The k-means algorithm for 2D images. This will generate an initial segmentation.
HMRf_EM.m	Function	The HMRf-EM algorithm.
MRF_MAP.m	Function	The MAP algorithm.
U_YX.m	Function	Calculating the likelihood energy.
U_X.m	Function	Calculating the prior energy.
U_l.m	Function	Calculating the clique potential.
gaussianBlur.m	Function	Blurring an image using Gaussian kernel.
gaussianMask.m	Function	Obtaining the mask of Gaussian kernel.
ind2ij.m	Function	Index to 2D image coordinates conversion.
G.m	Function	Calculating Gaussian distribution probability.
BoundMirrorExpand.m	Function	Expanding an image.
BoundMirrorShrink.m	Function	Shrinking an image.
Beijing World Park 8.JPG	Image	An example input image.

Table 1: Name and usage of each file in the *HMRf-EM-image* toolbox.

obtained by the k-means algorithm are not smooth enough, have morphological holes, and do not preserve the Canny edges. The HMRf refined labels overcome all these disadvantages.

6. Toolbox Documentation

We provide the name and usage of each file in our MATLAB toolbox *HMRf-EM-image* in Tabel 1. The `U_X.m` file can be modified to re-define pixel neighborhood relationships, and the `U_l.m` file can be modified to re-define the clique potentials. To reconfigure this toolbox for 3D image segmentation, the indexing system must be modified in several files.

7. Discussion

Our *HMRf-EM-image* toolbox is an implementation of the hidden Markov random field and its EM algorithm. This toolbox is well commented and easy to reconfigure. We have demonstrated the effectiveness of our toolbox on a simple example image. The HMRf model is mainly used to refine the direct segmentation output of some other algorithms. To get better segmentation results on more complicated images, some higher-level features should be used to construct \mathbf{y} instead of just pixel intensities, and some more advanced algorithm should be used to generate the initial labels.

References

- [1] J. Canny. A computational approach to edge detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions*

on, PAMI-8(6):679–698, Nov. 1986.

- [2] A. Saxena, S. Chung, and A. Ng. 3-d depth reconstruction from a single still image. *International Journal of Computer Vision*, 76:53–69, 2008.
- [3] N. M. Vaidya and K. L. Boyer. Discontinuity-preserving surface reconstruction using stochastic differential equations. *Computer Vision and Image Understanding*, 72(3):257–270, 1998.
- [4] L. Zhang and Q. Ji. Image segmentation with a unified graphical model. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(8):1406–1425, Aug. 2010.
- [5] Y. Zhang, M. Brady, and S. Smith. Segmentation of brain mr images through a hidden markov random field model and the expectation-maximization algorithm. *Medical Imaging, IEEE Transactions on*, 20(1):45–57, Jan. 2001.